

REMARKS

In the Office Action, the Examiner rejected claims 1-14 under 35 U.S.C. § 102(a) as being anticipated by U.S. Patent No. 5,987,247 to Lau (Lau). In this Amendment, Applicants amend claims 1-4, 7, 8, 12, and 14. The amendment of independent claims 1 and 7 are for clarification purposes only, and are not related to the Examiner's rejection or patentability. Likewise, the amendments to the dependent claims are for clarification purposes only, and are not related to the Examiner's rejection or patentability. For the convenience of the Examiner, the claims with brackets and underlining relative to the filed claims are in the APPENDIX.

The Examiner indicated that the IDS filed June 7, 1999, was not fully considered. Specifically, the Examiner did not consider the reference to the 1997 website. Pursuant to 37 C.F.R. § 1.98, a copy of the reference was provided along with a concise explanation of the relevance of the reference. Therefore, Applicants respectfully request that the Examiner consider the reference. Further, it is unclear what the Examiner means by “[the reference] should include screen shots and printouts of articles to be considered from the timeperiod.”

Claim 1 is directed to a “method for constructing a business application system by using a framework described by an object-oriented language.” Lau, however, does not disclose a method for constructing a business application system by using a framework constructed according the method of claim 1. Specifically, Lau does not disclose preparing an abstract class group including a system class group, a screen system class group, a report system class group, and a business logic system class group, as recited in claim 1.

LAW OFFICES
FINNEGAN, HENDERSON,
FARABOW, GARRETT,
& DUNNER, L.L.P.
1300 I STREET, N.W.
WASHINGTON, DC 20005
202-408-4000

In order to anticipate a claim, a reference must disclose each and every element of the claim. Because Lau does not disclose each element of claim 1, it cannot anticipate the claim. Thus, Applicants respectfully request that the Examiner reconsider and withdraw the rejection of claim 1 under § 102 as being anticipated by Lau.

Claim 7 is directed to a computer-readable storage medium having stored a framework for a business application system. As discussed above with respect to claim 1, Lau does not disclose a framework including an abstract class group including a system core class group, a screen system class group, a report system class group, and a business logic system class group, as recited in claim 7. Because Lau does not disclose each and every element of claim 7, it cannot anticipate the claim. Thus, Applicants respectfully request that the Examiner reconsider and withdraw the rejection of claim 7 under § 102 as anticipated by Lau.

Claim 11 is directed to a computer-readable storage medium having stored a framework for a business application system, and recites, among other things:

a system core class group having defined the manipulation of data;
and
a plurality of subclasses inheriting said system core class group.

Lau does not disclose a framework including a system core class group having defined the manipulation of data, and a plurality of subclasses inheriting said system core class group, as recited in claim 11. Because Lau does not disclose each and every element of claim 11, it cannot anticipate the claim. Applicants respectfully request that the Examiner reconsider and withdraw the rejection of claim 11 under § 102 as anticipated by Lau.

LAW OFFICES
FINNEGAN, HENDERSON,
FARABOW, GARRETT,
& DUNNER, L.L.P.
1300 I STREET, N.W.
WASHINGTON, DC 20005
202-408-4000

Claim 13 is directed to a computer-readable storage medium having a stored framework for a business application system, and recites, among other things:

a system core class group having defined the transmission and receiving of a request between functions; and
a plurality of subclasses inheriting said system core class group.

As discussed above with respect to claim 11, Lau does not disclose a framework including a system core class group having defined the transmission and receiving of a request between functions, and a plurality of subclasses inheriting the system core class group, as recited in claim 13. Because Lau does not disclose each and every element of claim 13, it cannot anticipate the claim. Thus, Applicants respectfully request that the Examiner reconsider and withdraw the rejection of claim 13 under § 102 as anticipated by claim 13.

Claims 2-6, 8-10, 12, and 14 depend from independent claims 1, 7, 11, or 13 and include all the limitations their respective independent claims. At least by virtue of their dependence, Applicants respectfully request that the Examiner reconsider and withdraw the rejection of claims 2-6, 8-10, 12, and 14 under § 102 as anticipated by Lau.

In view of the foregoing amendments and remarks, Applicants respectfully request the reconsideration and reexamination of this application and the timely allowance of the pending claims.

LAW OFFICES
FINNEGAN, HENDERSON,
FARABOW, GARRETT,
& DUNNER, L.L.P.
1300 I STREET, N.W.
WASHINGTON, DC 20005
202-408-4000

Please grant any extensions of time required to enter this response and charge any additional required fees to our deposit account 06-0916.

Respectfully submitted,

FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.

Dated: June 20, 2001

By: Kenneth M. Leich Reg. No.
Richard V. Burgujian
Reg. No. 31,744

LAW OFFICES
FINNEGAN, HENDERSON,
FARABOW, GARRETT,
& DUNNER, L.L.P.
1300 I STREET, N. W.
WASHINGTON, DC 20005
202-408-4000

APPENDIX

1. (Amended) A method for constructing a business application system by using a framework described by an object-oriented language, the method comprising the steps of:

preparing an abstract class group including (i) a system core class group, which has abstractly defined a basic structure and behavior of a business application system, and [an abstract class group, which inherits said system core class group and which includes] (ii) a screen system class group, a report system class group and a business logic system class group, which respectively inherit said system core class group;

inheriting said screen system class group, said report system class group and said business logic system class group of said abstract class group to prepare a screen system functional group, a report system functional group and a business logic system functional group;

inheriting said system core class group of said abstract class group to prepare a system core functional group; and

integrating said screen system functional group, said report system functional group, said business logic system functional group and said system core functional group.

2. (Amended) The method for constructing a business application system as set forth in claim 1, [wherein said] further comprising the step of preparing [step prepares, as said abstract class group, an abstract class group which further includes] a common component group including a plurality of common components commonly for use in said

LAW OFFICES
FINNEGAN, HENDERSON,
FARABOW, GARRETT,
& DUNNER, L.L.P.
1300 I STREET, N.W.
WASHINGTON, DC 20005
202-408-4000

business application system, each of said common components having an interface with said abstract class group.

3. (Amended) The method for constructing a business application system as set forth in claim 1, wherein [said preparing step prepares, as said abstract class group, an abstract class group including a plurality of abstract classes so that] each of said system core class group, said screen system class group, said report system class group and said business logic system class group [has] includes a plurality of abstract classes having a hierarchical structure based on at least one inheritance relationship.

4. (Amended) The method for constructing a business application system as set forth in claim 1, wherein [said preparing step prepares, as said abstract class group, an abstract class group formed so that] each of abstract classes included in each of said system core class group, said screen system class group, said report system class group and said business logic system class group includes an abstract method and a concrete method.

7. (Amended) A computer-readable storage medium having stored a framework for a business application system, which has been described by an object-oriented language, said framework including:

an abstract class group which has abstractly defined a structure and behavior of a business application system,

said abstract class group including (i) a system core class group, which has abstractly defined a basic structure and behavior of said business application system, and (ii) a screen system class group, a report system class group and a business logic system class group, which respectively inherit said system core class group.

LAW OFFICES
FINNEGAN, HENDERSON,
FARABOW, GARRETT,
& DUNNER, L.L.P.
1300 I STREET, N.W.
WASHINGTON, DC 20005
202-408-4000

8. (Amended) The computer-readable storage medium having stored a framework for a business application system as set forth in claim 7, [which] further [includes] including a common component group including a plurality of common components commonly for use in said business application system, each of said common components having an interface with said abstract class group.

12. (Amended) The computer-readable storage medium [The method for constructing a business application system] as set forth in claim 11, wherein said system core class group has defined the calling of a common component commonly for use in said business application system.

14. (Amended) The computer-readable storage medium [The method for constructing a business application system] as set forth in claim 13, wherein said system core class group has defined the calling of a common component commonly for use in said business application system.

LAW OFFICES
FINNEGAN, HENDERSON,
FARABOW, GARRETT,
& DUNNER, L.L.P.
1300 I STREET, N.W.
WASHINGTON, DC 20005
202-408-4000